IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re patent application of:

ANDREWS, Marcus J., et al.   Confirmation No.: **8059**

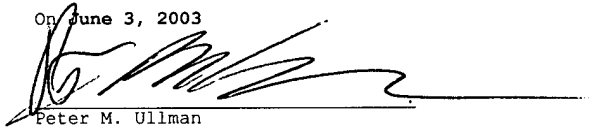Serial No.:   **09/497,914**   Group No.:   **2673**

Filed: **February 4, 2000**   Examiner: **Lewis, David Lee**

For: **GAME CONTROL DEVICE HAVING GENRE DATA**

```
I, Peter M. Ullman, Registration No. 43,963 certify
that this correspondence is being deposited with the
U.S. Postal Service as First Class mail in an
envelope addressed to the Commissioner for Patents,
P.O. Box 1450, Alexnadria, VA 22313-1450.

On June 3, 2003
                    Peter M. Ullman
                    Registration No: 43,963
```

Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

Sir:

RECEIVED

JUN 1 0 2003

Technology Center 2600

APPEAL BRIEF UNDER 37 C.F.R. § 1.192

This brief is in support of Appellant's appeal from the final rejection of claims 1-53 dated November 5, 2002. An advisory action was issued on January 27, 2003. A Notice of Appeal was filed on March 3, 2003.

A.   REAL PARTY IN INTEREST

The inventors in the present application have assigned their respective interests to MICROSOFT CORPORATION. An Assignment document was filed for recordation in the U.S. Patent and Trademark Office assignment system on May 30, 2000.

B.   RELATED APPEALS AND INTERFERENCES

There are no related appeals or interferences.

C.   STATUS OF CLAIMS

1. Claims 1-53 are pending. Claims 1-53 are reproduced in Appendix A, attached hereto. All claims (1-53) stand rejected as being unpatentable under 35 U.S.C. § 103(a). Claims 3, 14, 23, and 36 have been rejected as being unpatentable over U.S. Patent No. 5,442,376 to Tannenbaum, et al. ("Tannenbaum") in view of U.S. Patent No. 5,157,384 to Greanias, et al. ("Greanias"). Claims 4-13, 15-22, 24-31, and 37-53 have been rejected as being unpatentable over Tannenbaum and Greanias in view of U.S. Patent No. 6,085,265 to Kou ("Kou"). Claims 1, 2, and 32-35 have been rejected as being unpatentable over Kou in view of Tannenbaum and Greanias. Of the 53 pending claims, claims 1, 3, 14, 23, 32, 36, and 45 are independent.

D.   STATUS OF AMENDMENTS

No Amendments have been filed subsequent to the final rejection.

E.   SUMMARY OF THE INVENTION

The present invention is directed to a user input device that uses a structure called a "genre" to facilitate use of the device with certain types of software. The problem addressed by the invention is that input devices (e.g., pointing devices, joysticks, game steering wheels, etc.) generate data to be sent to their host device (e.g., a personal computer) when their controls are operated, and this data need to be interpreted by an application program in order to decide what action to take. For example, when a joystick is used to control a gaming application, the pushing of a button may indicate that the player wants to jump, or fire a weapon, depending upon what type of game is being played. Jumping and firing weapons are examples of "semantics." Conventionally, application software (e.g., a game) recognizes data generated by a particular input device;

since each device generates its own specific type of data to signify operation of the

controls on the device, software must normally be specifically configured to work with a

particular type of input device and to recognize the data that device generates. The

invention essentially allows an application program to be "decoupled" from the particular

vocabulary generated by a particular input device, and thus allows applications and input

devices to work together when they are not explicitly designed to do so.

The invention allows the various controls on a device to be linked with

semantics in a manner that is not dependent on the input lexicon of the application

program. The structure used to create this linkage is a "genre," which specifies various

different types of semantics for a class of application programs, and allows these

semantics to be linked with controls on the device.

As described in the present application, "[a] genre description for a

specific genre defines mappings between input device controls and actions to be

performed in a game of the genre. These actions are defined in terms of semantics or

labels."[1] Genres need not contain actual commands that can be fed as input to an

application program and interpreted by such a program. In fact, the Background section

of the present application describes the drawbacks of prior art systems that contain such

application-specific or device-specific commands:

> Some manufacturers of devices with new features provide
> filters to accommodate existing applications; frequently,
> these filters simulate keyboard presses or mouse
> movements for games that do not recognize enhanced
> features of the new device. However, these ad hoc
> approaches are error-prone, may result in a relationship
> between device controls and software actions that feels
> unnatural to the user, and can only provide support for
> applications the device manufacturer knows about and
> chooses to support.[2]

In other words, the present application describes the prior art as having the disadvantage

---

[1] Application, p. 9, ll. 20-23.

that it relies on direct linkage between the controls on a device and the input lexicon that

an application recognizes. (In the above example, it is assumed that "keyboard presses or

mouse movements" are the input that an application program understands.) Thus, the

present invention seeks to improve over the prior art by providing a way to connect input

devices with software in a way that is not dependent on the actual lexicon that an

application recognizes. The "Examples" section of the application[3] show examples of

genres that have semantics, but that do not include the command lexicon for a particular

application. Since the genre is not tied to the input vocabulary of any particular

application, it is possible to use the device with applications that are not specifically

designed for the device. The semantics in the genre are correlated with the actual input

lexicon of the software by an A-S Correlation (reference numeral 231).[4] The independent

claims, however, recite a genre, which is not the same as an A-S correlation.

In accordance with the application's description of a genre, the

independent claims recite that a genre comprises semantics but does not include

commands that are interpretable by the application program. As described in detail below

in section H, the claimed structure of a genre is different from the "user profile" taught in

Greanias and Tannenbaum, onto which the Examiner has read the claimed genre. The

prior art "user profile" correlates input messages with commands that an application

program understands. By contrast, the claimed genre does not include commands

interpretable by the application program. The absence of commands interpretable by an

application program is in sharp contrast to the prior art approach described above, since it

provides a mechanism whereby input devices and application programs may be used

together even when they are not specifically designed to do so. It should be noted that the

references cited by the Examiner merely describe this prior art approach, which

appellants distinguished in the Background section (as described above), and which is

---

[2] Application, p. 2, l. 20 through p. 3, l. 4.

[3] Application, pp. 24-31

[4] Application, FIG. 4

quite different from the claimed genre.

F.   ISSUE ON APPEAL

Whether claims 1-53 patentably define over Tannenbaum, Greanias, and Kou in the combinations proposed by the Examiner.

Inasmuch as all of the claim rejections rely on Tannenbaum, and Greanias (which is incorporated by reference into Tannenbaum) for the teaching of "a genre comprising a set of semantics and not including commands interpretable by the application program," the question on appeal is whether Tannenbaum and Greanias teach or suggest that claim feature, which is present in some form in all of the independent claims.

G.   GROUPING OF CLAIMS

For the purpose of this appeal, claims 1-53 stand or fall together.

H.   ARGUMENT

In the Office Action/Final Rejection dated November 5, 2002, the Examiner finally rejected claims 1-53 as being unpatentable over various combinations of Tannenbaum, Greanias, and Kou. For each of the independent claims, the Examiner relies on Tannenbaum (and Greanias, which is incorporated by reference into Tannenbaum) as teaching the feature of "a genre comprising a set of semantics and not including commands interpretable by the application programs." The Examiner's reliance on Tannenbaum and Greanias for this teaching is incorrect, since Tannenbaum and Greanias teach no such feature. The lynchpin of the section 103(a) rejection of the independent claims is the Examiner's assertion that Tannenbaum and Greanias teach the above-quoted feature. Since this assertion is not correct, the entire premise for the rejection of claims 1-53 fails, and the rejection must be reversed by the Board.

Independent claim 1 recites: "control mappings corresponding to a

plurality of program genres, the control mappings indicating actions to be performed in

application programs ..., *each of said genres comprising a set of semantics and not*

including commands interpretable by the application programs". All of the other

independent claims – 3, 14, 23, 32, 36, and 45 – recite features similar to the italicized

language – i.e., features to the effect that a genre comprises a set of semantics but does

not include commands interpretable by some class of application programs. The

Examiner asserts that Tannenbaum teaches this feature. In particular, the Examiner

specifically reads the claimed "genre" onto Tannenbaum's and Greanias's "user profile,"[5]

and then reasons as follows:

> [Regarding the limitation of] wherein each of said genres
> comprising a set of semantics and not including commands
> intereteble [sic] by the application program. Said
> limitation is also taught by Tannenbaum et al., wherein the
> user profile is interpreteble [sic] by the environmental link
> and not the application programs. The environmental link
> then sends the processed information to the application,
> Greanias et al., column 4 lines 14-27.[6]

<div align="center">* * *</div>

> Applicant [sic] amended the independant [sic] claims to
> include the limitation wherein each of said genres
> comprising a set of semantics and not including commands
> interpretable by the application programs. Said limitation is
> also taught by Tannenbaum et al., wherein the user profile
> is interpretable [sic] by the environmental link and not the
> application programs.[7]

In other words, the crux of the rejection is the Examiner's assertion that the claimed

"genre" reads on Tannenbaum's "user profile," and that Tannenbaum's user profile

does not include commands interpretable by the application programs.

---

[5] See Final Rejection dated 11/5/02, ¶¶ 3-6, 8, 11-12.
[6] See Final Rejection dated 11/5/02, ¶ 3.
[7] See Final Rejection dated 11/5/02, ¶ 13.

The problem with the Examiner's reasoning is that the premise on which it is based – i.e., that Tannenbaum's user profile "does not include commands interpretable by the application programs" – is simply false. Contrary to the Examiner's assertion, the "user profile" *does* contain commands that are interpretable by the application program. In particular, Tannenbaum states:

> The profiles [i.e., the user profile and the application
> profile] link the registration ID numbers of the input event
> to a corresponding command *which will be understood by*
> *the target object* or by another object which the user
> anticipates being in the system.[8]

Tannenbaum explains elsewhere that the "target object" can be an application program.[9] Thus, it is clear that Tannenbaum's "user profile" includes commands that are interpretable by an application program.

As noted above, Greanias is incorporated by reference into Tannenbaum, and Greanias also states quite clearly that the "user profile" includes commands that are interpretable by the application program. Specifically, Greanias states:

> The interface profile module 104 is comprised of
> sets of application profiles 105 and the user profiles 107,
> which are files which list input messages produced by the
> AIS 103 from the input signals received by input devices
> 36, 38, 40, mapped to keyboard, mouse or other commands
> *which are usable by* existing *application programs*, e.g.,
> mouse clicks, keystroke messages, MACROs, utility
> programs, etc.[10]

Based on the above quotation, it is simply beyond argument that the prior art "user profile" *does* contain commands interpretable by the application program. In view of the above quotations, the Examiner's assertion – i.e., that Tannenbaum and Greanias teaches a "user profile" that does not include commands interpretable by the application program – is simply not sustainable. Since the Examiner's rejection of each

---

[8] Tannenbaum, col. 8, ll. 18-21 (emphasis added).
[9] See Tannenbaum, col. 7, ll. 46-53.

claim is based on this incorrect assertion, the rejection of all of the claims must be reversed.[11]

It is clear from the foregoing that the rejection of the independent claims rests on the Examiner's incorrect assertion that the prior art teaches the exact opposite of what it actually teaches. This fact alone is sufficient to require reversal of the rejection. However, appellants submit that there are two additional reasons why the section 103(a) rejection of the independent claims is not sustainable.

First, it must be noted that a genre containing commands that are not interpretable by the application program is not merely an obvious innovation over the "user profile" of Tannenbaum and Greanias.[12] On the contrary, the claimed "genre" is substantially different from anything taught in the prior art, and there is nothing in the prior art of record that would motivate one of skill in the art to modify the user profile by removing from it the commands that are interpretable by the application programs. As explained extensively in Greanias, the commands contained in a user profile are passed directly to the application programs by an "environmental link."[13] In essence, when an input message is received, the environmental link determines whether there is a command corresponding to the message in the user profile. If such a command exists, in the user profile, then the environmental link passes the command directly to the

---

[10] Greanias, col. 7, l. 66 through col. 8, l. 5 (emphasis added).

[11] The Examiner reads the feature of a "genre" that does not include commands interpretable by the application program – which is substantially recited in each of the independent claims – onto Tannenbaum's and Greanias' "user profile." Since a dependent claim, by definition, incorporates the features of its base claim, each of the dependent claims is construed to recite this feature. Since the Examiner has not provided, for any of the dependent claims, an alternative construction of the claimed "genre," the rejection of the dependent claims effectively relies on the Examiner's assertion that the claimed "genre" is Tannenbaum's and Greanias' "user profile."

[12] It should be noted again that the Examiner has not asserted that the claimed "genre" is a modification of the prior art "user profile"; rather, the Examiner asserts that the claimed "genre" is the same as the prior art "user profile." However, as explained below, even if the Examiner were to acknowledge the difference between a "genre" and a "user profile," the prior art of record does not motivate any modification to the "user profile" that would yield that claimed "genre."

[13] Greanias, col. 8, l. 61 through col. 9, l. 7.

correct application program.[14] If the commands contained in the user profile were not interpretable by the application program, then the environmental link would not be able to perform its function. There is no teaching in any of the prior art cited – i.e., Greanias, Tannenbaum, or Kou – that suggests how the invention described in Greanias and Tannenbaum could be modified to exclude commands interpretable by an application program from the user profile.

Second, it should be noted that, even if such a modification could be made, the Examiner has not proposed any modification to the user profile of Greanias and Tannenbaum that would meet the claim limitation of a genre. It is the Examiner's burden to demonstrate how prior art references can be combined or modified to render a claimed invention obvious. See MPEP 706.02(j). The Examiner has not asserted that the prior art user profile can be modified to meet the claim feature of a "genre ... not including commands interpretable by the application programs." Instead, the Examiner has simply asserted that Greanias and Tannenbaum teach a user profile that has this feature. As demonstrated above, the Examiner's assertion is incorrect, and the Examiner has not met the burden of demonstrating that the claimed invention is obvious.

Finally, it should be noted that since each of the independent claims substantially calls for a genre that does not include commands interpretable by the application programs, the dependent claims are deemed to call for this feature by virtue of their dependency. 35 U.S.C. § 112, third paragraph. Since the independent claims are not obvious over the prior art of record, the dependent claims likewise are not obvious over the prior art of record.

For the reasons described above, the rejection of claims 1-53 is based on the faulty premise that Greanias's and Tannenbaum's "user profile" is a "genre" that does not include commands interpretable by an application program. Since this premise is

---

[14] *Id.*

incorrect – and since there is no other basis to conclude that claims 1-53 are obvious over the prior art of record – the rejection of claims 1-53 must be reversed.

I. CONCLUSION

       For all the foregoing reasons, appellants respectfully request that the Board reverse the rejection of claims 1-53, and that this application be returned to the Examiner for allowance.

Respectfully submitted,

Peter M. Ullman
Registration No. 43,963

Date: 6-3-03

WOODCOCK WASHBURN LLP
One Liberty Place - 46th Floor
Philadelphia, PA 19103
Tel: (215) 568-3100
Fax: (215) 568-3439

APPENDIX A

Claims on Appeal

1. A game control device that conforms to Universal Serial Bus (USB) device class definitions for Human Interface Devices (HIDs), comprising:

a plurality of human-actuated controls;

one or more HID descriptors that describe aspects of the human-actuated controls, the HID descriptors associating HID string indexes with the respective human-actuated controls;

control mappings corresponding to a plurality of application program genres, the control mappings indicating actions to be performed in application programs of particular genres in response to respective ones of the human-actuated controls, wherein the control mappings identify controls by their HID string indexes, each of said genres comprising a set of semantics and not including commands interpretable by the application programs.

2.   A game control device as recited in claim 1, the control mappings being indicated in data sets comprising:

a control section indicating the HID string indexes for the respective controls;

a genre section indicating actions to be performed in application programs of particular genres in response to respective ones of the human-actuated controls.

3. A computer peripheral comprising:

a plurality of human-actuated controls;

non-volatile memory containing control mappings corresponding to a plurality of application program genres, the control mappings indicating actions to be performed in application programs of particular genres in response to respective ones of the human-actuated controls, each of said genres comprising a set of semantics and not

including commands interpretable by the application programs.

4.   A computer peripheral as recited in claim 3, wherein the computer peripheral is a USB device and contains device class descriptions of the human-actuated controls in a format specified by the USB device class definition for human interface devices (HIDs), the control mappings containing references to HID identifiers for the respective human-actuated controls.

5.   A computer peripheral as recited in claim 3, wherein the computer peripheral is a USB device and contains descriptions of the human-actuated controls in a USB-specified format, the control mappings containing references to control identifiers contained in said descriptions.

6.   A computer peripheral as recited in claim 3, wherein the computer peripheral is a USB device and contains device class descriptions of the human-actuated controls in a format specified by the USB device class definition for human interface devices (HIDs), said device class definitions defining different HID string indexes for the respective human-actuated controls, the control mappings identifying controls by their different HID string indexes.

7.   A computer peripheral as recited in claim 3, the non-volatile memory containing a descriptor comprising:

         a control section indicating string indexes for the respective controls;

         a genre section indicating the control mappings for the respective application program genres.

8.   A computer peripheral as recited in claim 3, the non-volatile memory containing a descriptor comprising:

a control section indicating string indexes for the respective controls, the string indexes corresponding to separately defined human device interface (HID) string indexes;

a genre section indicating the control mappings for the respective application program genres, the control mappings identifying controls by their HID string indexes.

9.   A computer peripheral as recited in claim 3, the non-volatile memory containing a descriptor comprising:

a header section indicating the number of controls on the computer peripheral and the number of genres for which control mappings exist in the non-volatile memory;

a control section indicating string indexes for the respective controls;

a genre section indicating the control mappings for the respective application program genres;

a diagram section containing one more graphics images of the computer peripheral, the one or more graphics images identifying locations of the human-actuated controls on the computer peripheral.

10.   A computer peripheral as recited in claim 3, the non-volatile memory also containing control data that indicates:

string indexes for the respective controls;

graphics overlays that identify the human-actuated controls on the computer peripheral;

coordinates of the graphics overlays.

11.   A computer peripheral as recited in claim 3, the non-volatile memory also containing control data that indicates:

string indexes for the respective controls;

graphics overlays that identify the human-actuated controls on the

computer peripheral;

coordinates of the graphics overlays;

coordinates for pointers to the human-actuated controls.

12.   A computer peripheral as recited in claim 3, the non-volatile memory

containing a descriptor comprising:

a header section indicating the number of controls on the computer

peripheral and the number of genres for which control mappings exist in the non-volatile

memory;

a control section indicating string indexes for the respective controls, the

control section also indicating graphics overlays that identify the human-actuated controls

on the computer peripheral;

a genre section indicating the control mappings for the respective

application program genres;

13.   A computer peripheral as recited in claim 3, the non-volatile memory

further containing one more graphics images that identify the locations of the human-

actuated controls on the computer peripheral.

14. A method comprising:

defining a plurality of application program genres, each of said genres

comprising a set of semantics and not including commands interpretable by application

programs classifiable in the genre;

running an application program that has been classified as a particular

application program genre, wherein the application program is responsive to a plurality of

human-actuated controls on a control device;

querying the control device to obtain a genre descriptor, the genre descriptor indicating actions to be performed by an application program of said particular application program genre in response to respective ones of the human-actuated controls.

15.   A method as recited in claim 14, wherein the obtained genre descriptor comprises:

a control section indicating string indexes for the respective controls;

a genre section indicating the control mappings for the respective application program genres.

16.   A method as recited in claim 14, further comprising:

retrieving one or more HID descriptors from the control device, the HID descriptors describing aspects of the human-actuated controls, the HID descriptors associating HID string indexes with the respective human-actuated controls;

wherein the obtained genre descriptor identifies the human-actuated controls by their HID string indexes.

17.   A method as recited in claim 14, wherein the obtained genre descriptor comprises:

a control section indicating string indexes for the respective controls, the string indexes corresponding to separately defined human device interface (HID) string indexes;

a genre section indicating the control mappings for the respective application program genres, the control mappings identifying controls by their HID string indexes.

18.   A method as recited in claim 14, wherein the obtained genre descriptor comprises:

a header section indicating the number of controls on the control device

and the number of genres for which control mappings exist in the genre descriptor;

a control section indicating string indexes for the respective controls;

a genre section indicating the control mappings for the respective application program genres;

a diagram section containing one more graphics images of the control device, the one or more graphics images identifying locations of the human-actuated controls on the control device.

19.   A method as recited in claim 14, wherein the obtained genre descriptor comprises:

string indexes for the respective controls;

graphics overlays that identify the human-actuated controls on the control device;

coordinates of the graphics overlays.

20.   A method as recited in claim 14, wherein the obtained genre descriptor comprises:

string indexes for the respective controls;

graphics overlays that identify the human-actuated controls on the control device;

coordinates of the graphics overlays;

coordinates for pointers to the human-actuated controls.

21.   A method as recited in claim 14, wherein the obtained genre descriptor comprises:

a header section indicating the number of controls on the control device and the number of genres for which control mappings exist in the non-volatile memory;

a control section indicating string indexes for the respective controls, the

control section also indicating graphics overlays that identify the human-actuated controls on the control device;

a genre section indicating the control mappings for the respective application program genres;

22.   A method as recited in claim 14, wherein the obtained genre descriptor comprises one more graphics images that identify the locations of the human-actuated controls on the control device.

23. A computer-readable storage medium containing system services utilized by an application program to interact with a control device having a plurality of human-actuated controls, wherein the system services perform acts comprising:

receiving a request from an application program for a genre description corresponding to one of a plurality of application program genres, each of said genres comprising a set of semantics and not including commands interpretable by application programs classifiable in the genre;

querying the control device to obtain a genre descriptor, the genre descriptor indicating actions to be performed by an application program of said one of a plurality of application program genres in response to respective ones of the human-actuated controls;

returning the obtained genre descriptor to the requesting application program.

24.   A computer-readable storage medium as recited in claim 23, wherein the obtained genre descriptor comprises:

a control section indicating string indexes for the respective controls;

a genre section indicating the control mappings for the respective application program genres.

25.   A computer-readable storage medium as recited in claim 23, the systems services performs a further act comprising:

retrieving one or more HID descriptors from the control device, the HID descriptors describing aspects of the human-actuated controls, the HID descriptors associating HID string indexes with the respective human-actuated controls; wherein the obtained genre descriptor identifies the human-actuated controls by their HID string indexes.

26.   A computer-readable storage medium as recited in claim 23, wherein the obtained genre descriptor comprises:

a control section indicating string indexes for the respective controls, the string indexes corresponding to separately defined human device interface (HID) string indexes; a genre section indicating the control mappings for the respective application program genres, the control mappings identifying controls by their HID string indexes.

27.   A computer-readable storage medium as recited in claim 23, wherein the obtained genre descriptor comprises:

a header section indicating the number of controls on the control device and the number of genres for which control mappings exist in the genre descriptor;

a control section indicating string indexes for the respective controls;

a genre section indicating the control mappings for the respective application program genres;

a diagram section containing one more graphics images of the control device, the one or more graphics images identifying locations of the human-actuated controls on the control device.

28.   A computer-readable storage medium as recited in claim 23, wherein the obtained genre descriptor comprises:

string indexes for the respective controls;

graphics overlays that identify the human-actuated controls on the control device;

coordinates of the graphics overlays.

29.   A computer-readable storage medium as recited in claim 23, wherein the obtained genre descriptor comprises:

string indexes for the respective controls;

graphics overlays that identify the human-actuated controls on the control device;

coordinates of the graphics overlays;

coordinates for pointers to the human-actuated controls.

30.   A computer-readable storage medium as recited in claim 23, wherein the obtained genre descriptor comprises:

a header section indicating the number of controls on the control device and the number of genres for which control mappings exist in the non-volatile memory;

a control section indicating string indexes for the respective controls, the control section also indicating graphics overlays that identify the human-actuated controls on the control device;

a genre section indicating the control mappings for the respective application program genres;

31.   A computer-readable storage medium as recited in claim, wherein the obtained genre descriptor comprises one more graphics images that identify the locations of the human-actuated controls on the control device.

32. A data transmission medium carrying a data structure comprising:

a header section indicating the number of human-actuated controls on a computer peripheral and the number of application program genres for which control mappings exist in the data structure;

a control section indicating HID string indexes for the respective controls on the computer peripheral;

a genre section indicating control mappings for the respective application program genres, each of said genres comprising a set of semantics and not including commands interpretable by application programs classifiable in the genre.

33. A data transmission medium as recited in claim 32, further comprising:

a diagram section containing one more graphics images of the computer peripheral, the one or more graphics images identifying locations of the human-actuated controls on the computer peripheral.

34. A data transmission medium as recited in claim 32, wherein the control section also indicates graphics overlays that identify the human-actuated controls on the computer peripheral.

35. A data transmission medium as recited in claim 32, further comprising a diagram section, the diagram section comprising graphics overlays that identify the human-actuated controls on the computer peripheral;
wherein the control section indicates coordinates of the graphics overlays and coordinates for pointers to the human-actuated controls.

36. A method of using an input device connected to a computing device with software executable on said computing device, said method comprising the acts of:

running an application program which is responsive to input, said

application program being classified as a particular one of a plurality of application

program genres, each of said genres comprising a set of semantics and not including

commands interpretable by application programs classifiable in the genre;

querying a control device having a plurality of human-actuated controls,

said control device storing a genre descriptor indicating actions to be performed by

application programs in said particular application program genre in response to said

human-actuated controls;

obtaining, in response to said querying act, said genre descriptor; and

generating input to said application program in accordance with said genre

descriptor.

37. A method as recited in claim 36, wherein the obtained genre descriptor

comprises:

a control section indicating string indexes for the respective controls; and

a genre section indicating the control mappings for the respective

application program genres.

38. A method as recited in claim 36, further comprising:

retrieving one or more HID descriptors from the control device, the HID

descriptors describing aspects of the human-actuated controls, the HID descriptors

associating HID string indexes with the respective human-actuated controls;

wherein the obtained genre descriptor identifies the human-actuated controls by their HID

string indexes.

39. A method as recited in claim 36, wherein the obtained genre descriptor

comprises:

a control section indicating string indexes for the respective controls, the

string indexes corresponding to separately defined human device interface (HID) string

indexes; and

a genre section indicating the control mappings for the respective application program genres, the control mappings identifying controls by their HID string indexes.

40. A method as recited in claim 36, wherein the obtained genre descriptor comprises:

a header section indicating the number of controls on the control device and the number of genres for which control mappings exist in the genre descriptor;

a control section indicating string indexes for the respective controls;

a genre section indicating the control mappings for the respective application program genres; and

a diagram section containing one more graphics images of the control device, the one or more graphics images identifying locations of the human-actuated controls on the control device.

41. A method as recited in claim 36, wherein the obtained genre descriptor comprises:

string indexes for the respective controls;

graphics overlays that identify the human-actuated controls on the control device; and

coordinates of the graphics overlays.

42. A method as recited in claim 36, wherein the obtained genre descriptor comprises:

string indexes for the respective controls;

graphics overlays that identify the human-actuated controls on the control device;

coordinates of the graphics overlays;

coordinates for pointers to the human-actuated controls.

43. A method as recited in claim 36, wherein the obtained genre descriptor comprises:

a header section indicating the number of controls on the control device and the number of genres for which control mappings exist in the non-volatile memory;

a control section indicating string indexes for the respective controls, the control section also indicating graphics overlays that identify the human-actuated controls on the control device; and

a genre section indicating the control mappings for the respective application program genres.

44. A method as recited in claim 36, wherein the obtained genre descriptor comprises one more graphics images that identify the locations of the human-actuated controls on the control device.

45. A method of enabling the use of an application program that executes on a computing device with a control device having human-actuated controls, said method comprising the acts of:

defining a plurality of application program genres, each of said genres comprising a set of semantics and not including commands interpretable by application programs classifiable in the genre;

creating a genre descriptor, said genre descriptor indicating, for each one of said plurality of application program genres, actions to be performed by application programs in the respective application program genres in response to said human-actuated controls;

storing said genre descriptor in a memory of said control device, said

memory being communicatively coupleable to said computing device whereby said genre descriptor is accessible to said computing device.

46. A method as recited in claim 45, wherein the obtained genre descriptor comprises:

a control section indicating string indexes for the respective controls; and a genre section indicating the control mappings for the respective application program genres.

47. A method as recited in claim 45, further comprising:

retrieving one or more HID descriptors from the control device, the HID descriptors describing aspects of the human-actuated controls, the HID descriptors associating HID string indexes with the respective human-actuated controls; wherein the obtained genre descriptor identifies the human-actuated controls by their HID string indexes.

48. A method as recited in claim 45, wherein the obtained genre descriptor comprises:

a control section indicating string indexes for the respective controls, the string indexes corresponding to separately defined human device interface (HID) string indexes; and

a genre section indicating the control mappings for the respective application program genres, the control mappings identifying controls by their HID string indexes.

49. A method as recited in claim 45, wherein the obtained genre descriptor comprises:

a header section indicating the number of controls on the control device

and the number of genres for which control mappings exist in the genre descriptor;

a control section indicating string indexes for the respective controls;

a genre section indicating the control mappings for the respective application program genres; and

a diagram section containing one more graphics images of the control device, the one or more graphics images identifying locations of the human-actuated controls on the control device.

50. A method as recited in claim 45, wherein the obtained genre descriptor comprises:

string indexes for the respective controls;

graphics overlays that identify the human-actuated controls on the control device; and

coordinates of the graphics overlays.

51. A method as recited in claim 45, wherein the obtained genre descriptor comprises:

string indexes for the respective controls;

graphics overlays that identify the human-actuated controls on the control device;

coordinates of the graphics overlays;

coordinates for pointers to the human-actuated controls.

52. A method as recited in claim 45, wherein the obtained genre descriptor comprises:

a header section indicating the number of controls on the control device and the number of genres for which control mappings exist in the non-volatile memory;

a control section indicating string indexes for the respective controls, the

control section also indicating graphics overlays that identify the human-actuated controls
on the control device; and

a genre section indicating the control mappings for the respective
application program genres.

53. A method as recited in claim 45, wherein the obtained genre descriptor
comprises one more graphics images that identify the locations of the human-actuated
controls on the control device.